

## **REMARKS**

This is a full and timely response to the outstanding final Office Action mailed November 18, 2004. Reconsideration and allowance of the application and pending claims are respectfully requested.

### **I. Claim Rejections - 35 U.S.C. § 112, First Paragraph**

Claim 1 has been rejected under 35 U.S.C. § 112, first paragraph, for purportedly failing to comply with the enablement requirement in relation to the recitation “automatically determining without prompting from a user if an intercept library is enabled to process the event; if the intercept library is enabled to process the event, automatically transmitting . . .”

The enablement requirement requires that the specification describe the invention in such a manner as to enable one skilled in the art make and use the invention. MPEP § 2164. In making an enablement determination, the Examiner must determine whether the disclosure contained sufficient information regarding the subject matter of the claims as to enable one skilled in the art to make and use the invention. MPEP § 2164.01.

In the present case, there is sufficient information regarding the limitations “automatically” and “without prompting from a user” provided in Applicant’s disclosure so as to enable one skilled in the art to make and use the invention. For support of this contention, Applicant reproduces the following portion of Applicant’s specification (page 14, line 19 to page 15, line 11, emphasis added):

As shown, the application program 90 processes data and creates client events for service. *The application program 90 calls the API 110 to process the client event generated. The API 110 receives the request for service from the application program 90. The API 110 then determines if the intercept library 130 is enabled to handle events. If the API 110 determines that the intercept library 130 is not enabled to support the event, the API 110 processes the event and returns the event output to the application program 90.*

*If the API 110 determines that the intercept library 130 is enabled to support the event, the API 110 calls the intercept library 130 through the generic interception communication interface 120 to allow the intercept library 130 to process or handle the event. API 110 waits for a reply from the intercept library 130. If the intercept library 130 was able to process all event the API 110 waits to receive the next request for service of the application program 90. If the intercept library 130 was not able to process all the events, the API 110 processes the event and returns the event output to the application program 90.*

From the above excerpt, it is clear that no human intervention is required for the API 110 to determine if the intercept library 130 is enabled to handle events. To the contrary, that determination is described as one of the steps that the API 110 performs once a request is received. Accordingly, a person having ordinary skill in the art would readily appreciate that the API 110 makes the determination as to whether the intercept library is enabled “without prompting from a user”.

Given that the API 110 does not require human intervention to prompt it to make the determination about whether the intercept library is enabled, and further given that the API is a logic component that operates on a computing machine, it logically follows

that the API “automatically” determines if the intercept library is enabled to process the event.

In view of the above, it is clear that a person having ordinary skill in the art would be enabled to make and use the invention presently described in claim 1 from Applicant’s original disclosure. Applicant therefore respectfully requests that the rejection of claim 1 under 35 U.S.C. § 112, first paragraph, be withdrawn.

## **II. Claim Rejections - 35 U.S.C. § 112, Second Paragraph**

Claims 1-5 have been rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicant regards as the invention. In particular, the Examiner states that the limitation “generic interface communication interface” lacks antecedent basis.

In response to the rejection, Applicant has amended claim 1 to recite a “generic *interception* communication interface”, as recommended by the Examiner. In view of that amendment, it is respectfully asserted that claims 1-5 define the invention in the manner required by 35 U.S.C. § 112. Accordingly, Applicant respectfully requests that the rejections to those claims be withdrawn.

## **III. Claim Rejections - 35 U.S.C. § 102(b)**

Claims 1-5, 7-11, 13, 16-17, and 21-26 have been rejected under 35 U.S.C. § 102(b) as being anticipated by Smale (U.S. Pat. No. 5,764,985). Applicant respectfully traverses this rejection.

It is axiomatic that “[a]nticipation requires the disclosure in a single prior art reference of each element of the claim under consideration.” *W. L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1554, 220 USPQ 303, 313 (Fed. Cir. 1983).

Therefore, every claimed feature of the claimed invention must be represented in the applied reference to constitute a proper rejection under 35 U.S.C. § 102(b).

In the present case, not every feature of the claimed invention is represented in the Smale reference. Applicant discusses the Smale reference and Applicant's claims in the following.

**A. The Smale Disclosure**

The Office Action primarily relies upon lines 10-31 and 41-67 of column 4 of the Smale reference in rejecting Applicant's claims. Those sections of the Smale reference provide as follows:

In the WOSA configuration, application programs 23, 24 initiate function calls via an application program interface (API) 25 requesting the performance of various operations. From the API 25, the call is received by a routing component 26, which routes the requested function call through a service provider interface (SPI) 27 to one or more appropriate service providers. One of the service providers, e.g. service provider 28, may handle the request directly, or alternatively may call on other, lower-level service providers (not shown). Another service provider 29 is provided to call on operating system functions when so required by the requesting function call. . . .

FIG. 2A illustrates the general flow of operation when a typical function call is made from an application program. As previously described, with a layered architecture, a request router 30 of the routing component 26 initially receives the requesting call from the API 25, as indicated by step (1) of FIG. 2A. In keeping with the invention, at step (2) the function call is subsequently passed from the request router 30 to the notification manager 32. In turn, the notification manager 32 sequentially notifies each of the extensions 34, 35, at steps (3) and (4),

respectively, of the pending call before the call is made (pre-notification). This pre-notification step is also shown in FIG. 2B as step 16.

During this pre-notification procedure the extensions 34, 35 are called, enabling the extensions to execute their own procedures and perform operations according to their extended functionality thereby providing extensibility to the pending operation request. However, as described in more detail below, the extensions 34, 35 are limited in how they can ultimately affect the subsequent passing of the pending request to the appropriate service provider.

Following pre-notification of the extensions, the notification manager returns control of the call to the request router 30 at step (5), which then routes the call to one or more appropriate service providers, such as the service provider 28, at step (6). The service provider 28 services the requesting call, and when completed, returns control to the request router 30 at step (7). The call or calls to the service providers are also shown in FIG. 2B as step 18. It should be noted that an extension can also be a service provider itself. Accordingly, service providers may register as extensions with the notification manager 32.

Applicant claims methods and systems that are distinct from those described in the foregoing excerpt.

#### **B. Applicant's Claims 1-5 and 21-22**

Regarding independent claim 1, Smale does not teach or suggest “automatically determining without prompting from a user if an intercept library is enabled to process the event” or “if the intercept library is enabled to process the event, automatically transmitting the event from the application program interface to a generic interception communication interface having at least one intercept event send handler, the generic interception communication interface maintaining communication between the

application program interface and the intercept library”. First, no determination as to whether an “intercept library is enable to process the event” is made in the Smale system. There is simply no disclosure from Smale on this point. Second, no such determination is made before transmitting an event from an application program interface. Instead, as is evidenced by the above-provided excerpt, Smale simply discloses an application program interface (API) 25 that is configured for “requesting the performance of various operations.” Smale, column 4, lines 20-23. Nothing in the Smale disclosure suggests that the API 25 (or another component) makes the determination as to an intercept library’s readiness to process an event *before transmission from the API*. Third, Smale does not identify any “intercept library”. Instead, Smale only identifies “service providers” and “extensions.” Again, for a proper rejection under 35 U.S.C. § 102, the cited reference must disclose *each element* of the claim in question.

As a further matter, Applicant notes that Smale does not teach or suggest “transmitting the event from the generic interception communication interface to the intercept library with the at least one send handler”. First, as is mentioned above, Smale does not disclose an “intercept library”. Second, Smale does not disclose a “generic” interception communication interface. Although Smale describes a “routing component,” nowhere does Smale state that that component is generic. As is described in Applicant’s specification, the term “generic” refers to the ability of the interface to handle all API events without prior knowledge of the nature of those events. See, e.g., Applicant’s specification, page 7, lines 1-8. Third, Smale does not identify a “send handler” of the generic interception communication interface. Once again, a rejection under 35 U.S.C. § 102 is only proper if the cited reference discloses *each element* of the claim in question.

**C. Applicant's Claims 7-11 and 23-24**

Referring next to independent claim 7, Smale does not teach or suggest “means for determining whether any intercepting means is enabled to process the event” or “means for transmitting the event from the receiving and processing means to the intercepting means if the intercepting means is enabled to process the event”. First, as is mentioned above in relation to claim 1, Smale does not describe determining whether interception means is enabled to process an event, and therefore Smale fails to disclose means for performing that function. Second, Smale does not describe means for transmitting the event *if* the intercepting means is enable to process the event.

As a further matter, Smale does not teach or suggest “wherein the means for transmitting the event comprises a generic interception communication interface having at least one intercept event send handler, the generic interception communication interface maintaining communication between the intercepting means and the processing means” at least because Smale does not disclose a “generic” interception communication interface or an “event send handler”. Applicant refers the Examiner to the discussion of claim 1 above.

**D. Applicant's Claims 13, 16-17, and 25-26**

Regarding independent claim 13, Smale does not teach or suggest an application program interface that is configured to “determine if an intercept library is enabled to process the events and, if so, transmit the events to the intercept library”. As is noted above in relation to claim 1, Smale's API 25 is not described as comprising that functionality.

Smale further does not teach or suggest an “intercept library that is configured to process events”. Again, Smale does not disclose an intercept “library”.

Furthermore, Smale does not teach or suggest a “generic interception communication interface . . . having at least one intercept event send handler for maintaining communication between the application program interface and the intercept library”. As is mentioned in the foregoing, Smale neither discloses a “generic” interception communication interface nor an “event send handler”.

**E. Conclusion**

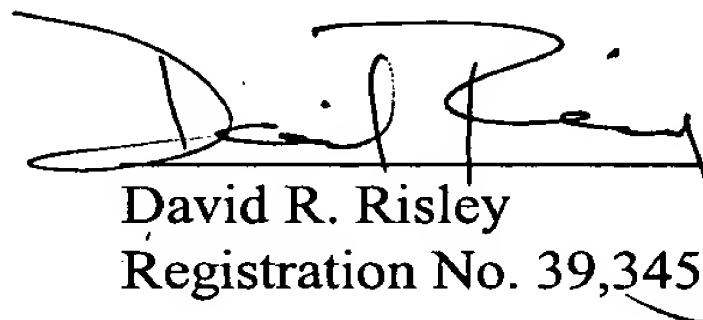
Due at least to the shortcomings of the Smale reference described in the foregoing, Applicant respectfully asserts that Smale does not anticipate Applicant's claims. Therefore, Applicant respectfully requests that the rejection of these claims be withdrawn.



### CONCLUSION

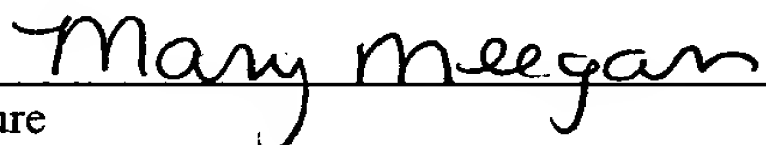
Applicant respectfully submits that Applicant's pending claims are in condition for allowance. Favorable reconsideration and allowance of the present application and all pending claims are hereby courteously requested. If, in the opinion of the Examiner, a telephonic conference would expedite the examination of this matter, the Examiner is invited to call the undersigned attorney at (770) 933-9500.

Respectfully submitted,

  
David R. Risley  
Registration No. 39,345

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail, postage prepaid, in an envelope addressed to: Assistant Commissioner for Patents, Alexandria, Virginia 22313-1450, on

1-5-05

  
Signature